

Simple Foreground Background Diff

Here we have a background picture and we would like to discover new objects on top of this background.

We will use the simple `opencv3.core` function **`absdiff`** and then apply a simple and large **`threshold`** to pick up all the new objects.

The reference clojure code file can be found here: [bgdiff.clj](https://github.com/opencv/opencv/blob/master/samples/cpp/bgdiff.clj)

Let's see first what our background looks like.

```
([ns scenic-iceberg
  (:require
   [opencv3.core :refer :all]
   [opencv3.utils :as u]))

(def bg (-> "resources/images/bgdiff/header.png" imread (u/resize-by 0.5)))
(u/mat-view bg)
```



Then put our hand in front of that background.

```
(def fg (-> "resources/images/bgdiff/front.png" imread (u/resize-by 0.5)))
(u/mat-view fg)
```



The **`absdiff`** function from OpenCV core is used, and we can directly see the output.

```
(def output (new-mat))
(absdiff bg fg output)
(u/mat-view output)
```



```
; diff in color
(def fg-1
  (-> output
   clone
   (threshold! 10 255 1)))
(u/mat-view fg-1)
```



You can then use a very permissive threshold to find out relevant shapes.

```
; diff in gray
(def fg-2
  (-> output
    clone
    (cvt-color! COLOR_RGB2GRAY)
    (threshold! 10 255 1)))
(u/mat-view fg-2)
```

